UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/837,036	04/18/2001	Tse-Hua Lan	US 010202	7522

24737 7590 03/01/2004

PHILIPS INTELLECTUAL PROPERTY & STANDARDS
P.O. BOX 3001
BRIARCLIFF MANOR, NY 10510

EXAMINER	
WONG, ALLEN C	
ART UNIT	**PAPER NUMBER**
2613	

DATE MAILED: 03/01/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/837,036	Applicant(s) LAN ET AL.
	Examiner Allen Wong	Art Unit 2613

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-15 and 22-24* is/are rejected.

7)☒ Claim(s) *16-21 and 25-30* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 4.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Information Disclosure Statement

1. The information disclosure statement (IDS) submitted on 9/9/02 is being considered by the examiner.

Claim Objections

2. Claim 4 is objected to because of the following informalities: line 1-2, "the computation load" is not previously defined. So, "the" needs to be replaced by "a" to provide antecedent basis to the term "computation load". Appropriate correction is required.

3. Claim 13 is objected to because of the following informalities: line 1, after "claim", "8" should be inserted instead of 1 because claim 1 is a method and claim 8 is a system. Appropriate correction is required.

4. Claim 14 is objected to because of the following informalities: on lines 15-16, the term "said encoded digital video signal" does not have antecedent basis, however, in line 1, if the applicant amends the term "data" of the term "an encoded data video signal", then that would provide proper antecedent basis for the term "said encoded digital video signal" on lines 15-16. Appropriate correction is required.

5. Claim 25 is objected to because of the following informalities: line 1, the term "total" should be included to provide antecedent basis. Appropriate correction is required.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors Protection Act of 1999 (AIPA) and the Intellectual Property and High Technology Technical Amendments Act of 2002 do not apply when the reference is a U.S. patent resulting directly or indirectly from an international application filed before November 29, 2000. Therefore, the prior art date of the reference is determined under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C. 102(e)).

7. Claims 1-3 and 7 are rejected under 35 U.S.C. 102(e) as being anticipated by Florencio (6,125,147).

Regarding claim 1, Florencio discloses an MPEG decoder for improving decoding efficiency by scaling the decoding of an encoded digital video signal (see fig.2 and col.2, ln.28-29), comprising:

means for decoding a compressed video data stream including a plurality of macroblocks, said decoding means being operative to output quantized data from said decoded data stream (col.6, ln.25-27 and fig.2, element 212 is a variable length decoder, VLD, that decodes compressed data and outputs quantized data);

means for performing an inverse quantization operation on said quantized data (col.6, ln.28-31 and fig.2, element 213);

means for performing an inverse discrete cosine transform (IDCT) of the output from said means for performing an inverse quantization operation (fig.2, element 214);

means for extracting header information from said quantized data from said decoding means and for performing a prediction operation according to predetermined criteria (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images);

means for generating a motion compensated reference value based on said quantized data from said decoding means (col.6, ln.40-45 and fig.2, element 216 is a motion compensator that generates a reference value based on the quantized data from the VLD 212); and,

means for adding the output from said IDCT means and said motion generating means to produce motion compensated pictures (col.6, ln.38-50; fig.2, element 215 is an adder that adds the output of the IDCT 214, S4, and the output of motion compensation means 216, S6, to produce motion compensated pictures S5).

Regarding claim 2, Florencio discloses the decoder of claim 1, further comprising means for storing the output of said adding means (fig.2, element 217 is the frame memory that stores the output of adder 215).

Regarding claim 3, Florencio discloses the decoder of claim 1, wherein said extracting means being operative to execute said prediction operation according to said extracted header information of the plurality said macroblocks (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information that contains the details of data from the macroblocks, to properly decode the information in preparation for reconstruction and display of images).

Regarding claim 7, Florencio discloses the decoder of claim 1, wherein said prediction operation is implemented using a processing unit and software which controls the operation of said processing unit (col.7, ln.9-25; fig.2, element 230 is a controller that uses the header data to implement a prediction operation using a processing unit 234 and software for properly processing the decoding of image data).

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 4, 5, 8-11, 13-15 and 22-23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Florencio (6,125,147) in view of Tan (6,408,096).

Regarding claim 4, Florencio discloses the extraction of the header information from the quantized data from the decoding means and for performing a prediction operation according to predetermined criteria (col.6, ln.62 to col.7, ln.2-8; the variable

length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images). Florencio does not specifically disclose wherein said prediction operation defines the computation load of said IDCT means and said motion generating means.

However, Tan teaches that the prediction operation defines the computation load of the IDCT means and the motion generating means (col.8, ln.23-60; in fig.1, element 180 has the complexity estimator is applied to the decoder 190 and a feedback loop is used to recursively estimate the complexity or to *selectively adjust* or *predict* the complexity, ie. the computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the computational load of the MPEG decoder is *selectively adjusted*).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 5, Florencio discloses the extraction of the header information from the quantized data from the decoding means and for performing a prediction operation according to predetermined criteria (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images). Florencio does not specifically disclose wherein the computation load of said MPEG decoder is selectively adjusted by scaling said IDCT means and said motion generating means based on said prediction operation.

However, Tan teaches the computation load of the MPEG decoder is selectively adjusted by scaling the IDCT means and the motion generating means based on the prediction operation (col.1, ln.29-30, MPEG compression/decompression is used; col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate the complexity or to *selectively adjust* or *predict* the complexity, ie. the computational load of the IDCT and motion generating

means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the computational load of the MPEG decoder is *selectively adjusted*).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 8, Florencio discloses a programmable video decoding system (col.7, ln.9-25; fig.2, element 230 is a controller that uses the header data to implement a prediction operation using a processing unit 234 and software for properly processing or programming the decoding of image data), comprising:

a variable length decoder (VLD) configured to receive and decode a stream of block-based data packets, said VLD being operative to output quantized data from said

decoded data packets (col.6, ln.25-27 and fig.2, element 212 is a variable length decoder, VLD, that decodes compressed data and outputs quantized data);

a variable length decoder configured to extract the header information from said block-based data packets (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images);

an inverse quantizer coupled to receive the output of said variable length decoder to operatively inverse quantize the quantized data received from said variable length decoder (col.6, ln.28-31 and fig.2, element 213);

an inverse discrete cosine transformer (IDCT) coupled to the output of said inverse quantizer for transforming the dequantized data from frequency domain to spatial domain (fig.2, element 214);

a motion compensator (MC) configured to receive motion vector data from said quantized data and to generate a reference signal (col.6, ln.40-45 and fig.2, element 216 is a motion compensator that generates a reference signal S6 based on the quantized data from the VLD 212); and,

an adder for receiving said reference signal and said spatial domain data from said IDCT to form motion compensated pictures (col.6, ln.38-50; fig.2, element 215 is

an adder that adds the output of the IDCT 214, S4, and the output of motion compensation means 216, S6, to produce motion compensated pictures S5).

Florencio does not specifically disclose a complexity estimator configured to extract the header information from said block-based data packets and further configured to execute a video complexity algorithms based on said extracted header information. However, Tan teaches a complexity estimator configured to extract the header information from the block-based data packets and further configured to execute a video complexity algorithms based on the extracted header information (col.8, ln.23-32; in fig.1, at the encoding side, the complexity parameters are inserted into the header information section of the block-based data packets, and then, at the decoding side, the complexity parameter information in the header is sent to the complexity estimator 180 in order to extract the header information with the complexity parameter data in preparation for affecting, scaling or *configuring* the decoder 190 to properly decode the video images by executing video complexity algorithms based on the extracted header information, also in col.1, ln.18-21 and col.8, ln.13-22, Tan teaches the implementation of software decoding or video complexity algorithms for decoding).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 9, Florencio discloses a buffer configured to store the output of the adder (fig.2, element 217 is the frame memory that stores the output of adder 215).

Regarding claim 10, Florencio discloses the variable length decoder configured to extract the header information from the block-based data packets (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data in the block-based data packets originating from the video stream data). Florencio does not specifically disclose wherein said complexity estimator being operative to execute said video complexity algorithms according to the header information of said decoded block-based data packets, and wherein the computation load of said decoding system is regulated by scaling said IDCT and said MC based on said video complexity algorithms.

However, Tan teaches the complexity estimator being operative to execute the video complexity algorithms according to the header information of the decoded block-based data packets (col.8, ln.23-32; in fig.1, at the encoding side, the complexity parameters are inserted into the header information section of the block-based data packets, and then, at the decoding side, the complexity parameter information in the header is sent to the complexity estimator 180 in order to extract the header information with the complexity parameter data in preparation for affecting, scaling or *configuring* the decoder 190 to properly decode the video images by executing video complexity algorithms according to the extracted header information of the decoded block-based data packets, also in col.1, ln.18-21 and col.8, ln.13-22, Tan teaches the implementation of software decoding or video complexity algorithms for decoding), and

wherein the computation load of the decoding system is regulated by scaling the IDCT and the MC based on the video complexity algorithms (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate the complexity or to *regulate* the complexity, ie. the computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the computational load of the decoding system is *regulated*, where also in col.1, ln.18-21 and col.8, ln.13-22, Tan teaches the implementation of software decoding or video complexity algorithms for decoding).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 11, Florencio discloses the variable length decoder configured to extract the header information from the block-based data packets (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data in the block-based data packets originating from the video stream data). Florencio does not specifically disclose the computation load of said IDCT and said MC are selectively adjusted according to said video complexity algorithms.

However, Tan teaches the computation load of said IDCT and said MC are selectively adjusted according to said video complexity algorithms (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate the complexity or to *selectively adjust* the complexity, ie. the computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *selectively adjusted*, so the IDCT 194 is *scaled or selectively adjusted* and the motion generating means 196 is *scaled or selectively adjusted*, thus, the computational load of the

decoding system is *selectively adjusted* based on the video , where also in col.1, ln.18-21 and col.8, ln.13-22, Tan teaches the implementation of software decoding or video complexity algorithms for decoding).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 13, Florencio discloses the system where decoding is implemented using a processing unit and software, which controls the operation of the processing unit (col.7, ln.9-25; fig.2, element 230 is a controller that uses the header data to implement a prediction operation using a processing unit 234 and software for properly processing the decoding of image data). Florencio does not specifically disclose the video complexity algorithm is implemented using a processing unit and software, which controls the operation of the processing unit. However, Tan teaches the video complexity algorithm is implemented using a processing unit and software, which controls the operation of the processing unit (in col.1, ln.18-21 and col.8, ln.13-22, Tan teaches the implementation of software decoding or video complexity algorithms for decoding, and that since Tan's decoding operation clearly occurs on a computer with the use of software and hardware, Tan must disclose a processing unit in the computer hardware otherwise the software would not run without the hardware components of the computer).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 14, Florencio discloses a method for improving decoding efficiency of an encoded data video signal employing an MPEG digital video decoder (see fig.2 and col.2, ln.28-29) having a variable length code decoder (fig.2, element 212), an inverse quantizer (fig.2, element 213), an inverse discrete cosine transformer (fig.2, element 214), and a motion compensator (fig.2, element 216), said method comprising the steps of:

receiving a compressed video data stream at said VLC decoder and producing decoded data therefrom (col.6, ln.25-27 and fig.2, element 212 is a variable length decoder, VLD, that receives the compressed video data stream, decodes compressed data and outputs quantized data);

retrieving header information from said decoded data (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images);

dequantizing said decoded data using said inverse quantizer(IQ) to generate dequantized, decoded data (col.6, ln.28-31 and fig.2, element 213);

employing said IDCT for transforming said dequantized, decoded data from frequency domain to spatial domain to produce difference data (fig.2, element 214 produces difference data S4);

employing said MC for generating a reference data of said encoded digital video signal (col.6, ln.40-45 and fig.2, element 216 is a motion compensator that generates a reference data S6 based on the quantized data from the VLD 212); and,

combining said reference data and said difference data to produce motion compensated pictures (col.6, ln.38-50; fig.2, element 215 is an adder that adds or combines the difference data S4 and the reference data S6 to produce motion compensated pictures S5).

Florencio does not specifically disclose an estimator for calculating a total computation load for said IDCT and said MC based on the classification of said header information. However, Tan teaches a complexity estimator for calculating a total computation load for the IDCT and the MC based on the classification of said header information (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate the complexity or to *calculate* the complexity, ie. the total computation load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal

184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the computational load of the decoding method and system is *calculated*; and also in col.8, ln.23-32, fig.1, at the encoding side, the complexity parameters are inserted into the header information section of the block-based data packets, and then, at the decoding side, the complexity parameter information in the header is sent to the complexity estimator 180 in order to extract the header information with the complexity parameter data so that *calculation* of the complexity or *total computation load* is done in preparation for affecting, scaling or *configuring* the decoder 190 to properly decode the video images according to the extracted header information of the decoded macroblock-based data packets).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 15, Florencio discloses retrieving header information from the decoded data (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in

decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images). Florencio does not specifically disclose comprising the step of transmitting said total computation load to said IDCT and said MC.

However, Tan teaches the step of transmitting the total computation load to the IDCT and the MC (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate or calculate the complexity, ie. the total computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information or the *total computation load* is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the total computational load is *transmitted* to the IDCT and the MC).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 22, Florencio discloses a prediction method of improving decoding efficiency of an encoded data video signal employing an MPEG digital video decoding system (fig.2 and col.2, ln.28-29) having a variable length code decoder (fig.2, element 212), an inverse quantizer (fig.2, element 213), an inverse discrete cosine transformer (fig.2, element 214), a motion compensator (fig.2, element 216), and an estimator, said method comprising the steps of:

decoding a compressed bitstream including a plurality of macroblocks to obtain a corresponding decoded macro block (col.6, ln.25-27 and fig.2, element 212 is a variable length decoder, VLD, that receives the compressed video data stream, decodes compressed data and outputs quantized data and decoded macroblock); and

obtaining a header classification criteria from the header information of said decoded macroblock (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details or *header classification criteria* in video sequence data, Group of Picture (GOP) data, picture data, and macroblock data, to properly decode the information in preparation for reconstruction and display of images).

Florencio does not specifically disclose an estimator employing said estimator for predicting a total computation load according to the header information from said decoded macroblock and forwarding said total computation load to said IDCT and said MC; and, adjusting the computation load of said IDCT and said MC according to said total computation load.

However, Tan teaches an estimator employing the estimator (fig.1, element 180 where in fig.6, the specifics of element 180 is shown in that element 183 is a complexity estimator) for predicting a total computation load according to the header information from the decoded macroblock (col.8, ln.23-32; in fig.1, at the encoding side, the complexity parameters are inserted into the header information section of the macroblock based data packets, and then, at the decoding side, the complexity parameter information in the header is sent to the complexity estimator 180 in order to extract the header information with the complexity parameter data so that *calculation* of the complexity or *total computation load* is done in preparation for affecting, scaling or *configuring* the decoder 190 to properly decode the video images) and

forwarding the total computation load to the IDCT and the MC (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate or calculate the complexity, ie. the total computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and

that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation of the complexity or the computational load of the IDCT and the motion generating means, and the complexity information or the *total computation load* is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the total computational load is *forwarded* to the IDCT and the MC); and,

adjusting the computation load of the IDCT and the MC according to the total computation load (col.8, ln.23-60 and fig.1, element 180, the complexity estimator is applied to the decoder 190 and a feedback loop is used to further estimate or calculate the complexity, ie. the total computational load of the IDCT and motion generating means, of the next decoded picture, wherein fig.6, the specifics of the complexity estimator are further disclosed in that the error 186 of the complexity measure is used by calculating the difference between the estimated complexity signal 184 and the actual complexity signal 185, and that the error 186 goes back to the feedback gain 187, and then the output of the feedback gain eventually goes back to the complexity estimator 183 for constant recalculation or re-*adjustment* of the complexity or *the computational load of the IDCT and the motion generating means (MC)*, and the complexity information or the *total computation load* is implemented into decoder 190 of fig.2 where the components of the decoder 190 are affected or *scaled*, so the IDCT 194 is *scaled* and the motion generating means 196 is *scaled*, thus, the computation load of the IDCT and MC is *adjusted* accordingly to the total computational load).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

Regarding claim 23, Florencio teaches the decoding of the decoded macroblock according to the header information (col.6, ln.62 to col.7, ln.8; note VLD 212 decodes the decoded macroblock according to the header information and sends the fidelity indicative parameter "DATA" where "DATA" contains "other information" regarding the video frames) together with motion vector information received in the compressed bitstream (col.6, ln.25-27, in fig.2, VLD 212 has the motion vector information received in the compressed bitstream, as evidenced by the exit of MV or motion vectors from element 212). Florencio does not specifically disclose further comprising the step of decoding the decoded macroblock according to the total computation load, together with motion vector information received in the compressed bitstream.

However, Tan teaches the step of decoding the decoded macroblock according to the total computation load, together with motion vector information received in the compressed bitstream (col.8, ln.13-22 and in fig.1, note the decoder 190, the macroblock data is decoded according to the complexity or the total computational load as estimated by complexity estimator 180, and that the decoder 190 has a VLD 191, in fig.2, that receives the motion vector information from the compressed bitstream stored at the buffer 170, and so the decoder 190 uses the total computation load, calculated

by complexity estimator 180, together with the motion vector information received in the compressed bitstream).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Tan's complexity estimator with Florencio's video decoder method and system for properly decoding as much video data as possible without losing pertinent information, and for versatile decoding in multiple formats and resolutions (col.8, ln.63 to col.9, ln.4).

10. Claim 6 is rejected under 35 U.S.C. 103(a) as being unpatentable over Florencio (6,125,147) in view of Wittig (6,011,498).

Regarding claim 6, Florencio discloses the extraction of the macroblock header information (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details of the macroblock data, to properly decode the information in preparation for reconstruction and display of images). Florencio does not specifically disclose wherein said extracted header information includes a macroblock-type, a motion vector magnitude, a motion vector count, non-zero discrete cosine transformer(DCT) coefficients, and a coded block pattern (CBP) number from said decoded block-based data packets.

However, this limitation is well known to one of ordinary skill in the art, but if one is not convinced, then one can peruse Wittig since Wittig teaches the extracted header information includes a macroblock-type, a motion vector magnitude, a motion vector count, non-zero discrete cosine transformer(DCT) coefficients, and a coded block

pattern (CBP) number from said decoded block-based data packets (col.3, ln.9-14; Wittig discloses the header information contains macroblock type, motion type and a plurality of motion vectors which includes motion vector magnitude and count, DCT type which includes non-discrete cosine transformer coefficients, and a coded block pattern number).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Wittig's teaching of utilizing the header information with Florencio's video decoding method and system for obtaining relevant header information so as to decode MPEG video data streams more efficiently and quickly, and at the same time, preserving high image quality (col.1, ln.44-50).

11. Claims 12 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Florencio (6,125,147) and Tan (6,408,096) as applied to claims 8 and 22 above, and further in view of Wittig (6,011,498).

Regarding claims 12 and 24, Florencio discloses the extraction of the macroblock header information (col.6, ln.62 to col.7, ln.2-8; the variable length decoder (VLD) 212 in decoder 210 extracts the header information from the quantized data and performs a prediction operation, by using this header information which contains details of the macroblock data, to properly decode the information in preparation for reconstruction and display of images). Also, Tan discloses the extraction of header information (col.8, ln.23-32). Florencio and Tan do not specifically disclose wherein said extracted header information includes a macroblock-type, a motion vector magnitude, a motion vector

count, non-zero discrete cosine transformer(DCT) coefficients, and a coded block pattern (CBP) number from said decoded block-based data packets.

However, this limitation is well known to one of ordinary skill in the art, but if one is not convinced, then one can peruse Wittig since Wittig teaches the extracted header information includes a macroblock-type, a motion vector magnitude, a motion vector count, non-zero discrete cosine transformer(DCT) coefficients, and a coded block pattern (CBP) number from said decoded block-based data packets (col.3, ln.9-14; Wittig discloses the header information contains macroblock type, motion type and a plurality of motion vectors which includes motion vector magnitude and count, DCT type which includes non-discrete cosine transformer coefficients, and a coded block pattern number).

Therefore, it would have been obvious to one of ordinary skill in the art to combine Wittig's teaching of utilizing the header information with Florencio and Tan's combined video decoding method and system for obtaining relevant header information so as to decode MPEG video data streams more efficiently and quickly, and at the same time, preserving high image quality (col.1, ln.44-50).

Allowable Subject Matter

12. Claims 16-21 and 25-30 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

13. The following is a statement of reasons for the indication of allowable subject matter: Florencio discloses a method and apparatus for reducing breathing artifacts in

compressed video. Tan discloses a method for computational graceful degradation in an audiovisual compression system. Wittig discloses a generic dual-speed variable length decoding architecture for MPEG-2 video data. Neither Florencio, Tan, Wittig, nor any of the prior art does not disclose details or specifics of the combination of limitations as disclosed in claims 16 and 25: wherein the step of calculating and predicting said total computation load further comprising the steps of: determining a first computation load based on the macroblock of said retrieved header information; determining a second computation load based on the motion magnitude of said retrieved header information; determining a third computation load based on the motion vector magnitude of said retrieved header information; determining a fourth computation load based on the number of non-zero DCT coefficients of said retrieved header information; determining a fifth computation load based on the coded block pattern number of said retrieved header information; and, combining said first, second, third, fourth, fifth computation loads, and an average computation load to obtain said total computation load. Dependent claims 17-21 and 26-30 contain allowable subject matter for the same reasons as claims 16 and 25.

Conclusion

14. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Siracusa (US 5,289,276) discloses a method and apparatus for conveying compressed video data over a noisy communication channel.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Allen Wong whose telephone number is (703) 306-5978. The examiner can normally be reached on Mondays to Thursdays from 8am-6pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Christopher Kelley can be reached on (703) 305-4856. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Allen Wong
Examiner
Art Unit 2613

AW
2/22/04